STUDENT REPORT

DETAILS

Name

ZIHAN

Roll Number

3BR23EE113

EXPERIMENT

Title

SUM OF NUMBERS AT PRIME FACTORS

Description

Prime factors of a positive integer are the prime numbers that divide that integer exactly.

Given an array arr of n integers and a positive integer num.

Let's suppose prime factorization of num is: $p^a \times q^b \times r^c \times .... \times z^f$,where p,q,r...z are prime numbers.

Sum of numbers in array arr at indices of prime factors of number num is: a x arr[p] + b x arr[q] + c x arr[r] +..... + f x arr[z].

You are given an array arr of size n and a positive integer num. You are required to calculate the sum of numbers in arr as mentioned above, and print the same.

Note:

- If arr is empty, print -1.
- If prime factor of num not found as indices, print 0.

Input Format:

The input consists of three lines:

- The first line contains an integer, i.e. n.
- The second line contains an array arr of length of n.
- The third line contains an integer num

The input will be read from the STDIN by the candidates.

Output Format:

Print the sum that was mentioned in the problem statement.

Example:

Input:

6

11 21 32 45 1 23

6

Output:

77

Explanation:

$6=2^1 \times 3^1$

Source Code:

```
def prime_factors(n):
    """Return a list of prime factors of n."""
    factors = []
    # Check for number of 2s that divide n
    while n % 2 == 0:
        if 2 not in factors:
            factors.append(2)
        n //= 2

    # n must be odd at this point, check for odd facto
rs from 3 to sqrt(n)
    for i in range(3, int(n**0.5) + 1, 2):
        while n % i == 0:
            if i not in factors:
                factors.append(i)
            n //= i

    # This condition is to check if n is a prime numbe
r greater than 2
    if n > 2:
        factors.append(n)

    return factors

def calculate_sum(arr, num):
    if not arr:
        return -1  # Handle empty array case

    # Get the prime factors of num
    factors = prime_factors(num)

    # Calculate the sum based on indices from the prim
e factors
    total_sum = 0
    for factor in factors:
        if factor < len(arr):  # Check if the index is
within bounds
            total_sum += arr[factor]

    return total_sum

# Input reading
n = int(input().strip())
arr = list(map(int, input().strip().split()))
num = int(input().strip())

# Output the result
result = calculate_sum(arr, num)
print(result)
```

sum=1*arr[2]+1*arr[3]=1*32+1*45=77

RESULT

2 / 5 Test Cases Passed | 40 %